

IN THE CLAIMS:

Please find below a listing of all pending claims. The statuses of the claims are set forth in parentheses. For those currently amended claims, underlined emphasis indicates insertions and ~~strike through~~ emphasis (and/or double brackets) indicates deletions.

1. (cancelled)
2. (previously presented) The method according to claim 6, wherein the LU decomposition is executed in parallel by each processor of each node in a recursive procedure.
3. (currently amended) The method according to claim 6, wherein
in said update step, while computing part of a block of the matrix which is not yet LU-decomposed ~~a row block~~, each node transfers data that belongs to a computed block and is needed to update other blocks, to other nodes in parallel to the computation.
4. (previously presented) The method according to claim 6, wherein
said parallel computer is a SMP node distributed-memory type parallel computer in which each node is a SMP (symmetric multi-processor).
5. (currently amended) A matrix processing device of a parallel computer in which ~~a plurality of processors and~~ a plurality of nodes, each including memory a memory and a processor are connected through a network, wherein the plurality of nodes includes n nodes, comprising:
a first allocation unit ~~dividing to divide~~ an array $A(1:k, 1:k)$ of a matrix to be processed by the number n of nodes to create n divided matrices and ~~assigning to assign~~ divided matrices subarrays $A(1:k/n, 1:k), \dots, A(k(n-1)/n:k, 1:k)$ to the divided matrices, ~~dividing to divide~~ one of the subarrays into a narrow block by an integer

~~m, and wherein each node's reading node is to read~~ data from memory so that a first narrow block is placed in a first node of the plurality of nodes, a second narrow block is placed in a second node of the plurality of nodes, ... , a m-th narrow block is placed in a $(\text{mod}(m-1,n)+1)$ -th node of the plurality of nodes, wherein $\text{mod}(m-1,n)$ is modulo of m-1 divided by n;

a separation unit eliminating to eliminate data corresponding to a diagonal blocks block $A(\text{nbase}:\text{nbase}+m, \text{nbase}:\text{nbase}+m)$, where nbase being a position of an upperleft element of the diagonal block and m are being integers integers, from data of the narrow blocks placed in each node, at each node;

a second allocation unit to redundantly allocating allocate same data as the diagonal block which is eliminated at each node to each node commonly;

an LU decomposition unit applying to apply LU decomposition to both the diagonal block and the an allocated block by the first allocation unit in parallel in each node; and

an update unit updating to update the block of the matrix which is not yet LU-decomposed, using an LU-decomposed block, at each node~~[.].~~

thereby realizing fast LU decomposition of a matrix effectively using a hardware of the parallel computer.

6. (currently amended)A matrix processing method of a parallel computer in which a ~~plurality of processors and~~ a plurality of n nodes ~~each including memory a memory and a processor~~ are connected through a network, comprising:

dividing an array $A(1:k, 1:k)$ of a matrix to be processed by the number n of nodes and assigning ~~divided matrices~~ subarrays $A(1:k/n, 1:k)$, ... , $A((k(n-1)/n):k, 1:k)$ to divided matrices, dividing one of the subarrays into narrow block by integer m, and each node's reading data from memory of each node so that a first narrow block is placed in a first node, a second narrow block is placed in a second node, ... , a m-th narrow block is placed in a $(\text{mod}(m-1,n)+1)$ -th node, wherein $\text{mod}(m-1,n)$ is modulo of m-1 divided by n;

eliminating data corresponding to a diagonal blocks block $A(\text{nbase}:\text{nbase}+\text{m}, \text{nbase}:\text{nbase}+\text{m})$, where nbase being a position of an upperleft element of the diagonal block and m are integers integers, from data of the narrow blocks placed in each node, at each node;

redundantly allocating same data as the diagonal block which is eliminated at each node to each node commonly;

applying LU decomposition to both the diagonal block and ~~the~~ an allocated block by the assigning step in parallel in each node; and

updating the blocks of the matrix which is not yet LU-decomposed, using an LU-decomposed block, at each node $[[.]]$.

thereby realizing fast LU decomposition of a matrix effectively using a hardware of the parallel computer.

7. (currently amended)A computer-readable storage medium on which is recorded a program for enabling a computer to realize a matrix processing method as a parallel computer in which ~~a plurality of processors and~~ a plurality of nodes each including ~~memory a memory and a processor~~ are connected through a network, the method comprising:

dividing an array $A(1:k, 1:k)$ of a matrix to be processed by the number n of nodes to create divided matrices and assigning ~~divided matrices~~ subarrays $A(1:k/n, 1:k), \dots, A(k(n-1)/n+1:k, 1:k)$ to divided matrices, dividing one of the subarrays into narrow block by integer m , and each node's reading data from memory of each node so that a first narrow block is placed in a first node, a second narrow block is placed in a second node, \dots , a m -th narrow block is placed in a $(\text{mod}(m-1, n)+1)$ -th node, wherein $\text{mod}(m-1, n)$ is modulo of $m-1$ divided by n ;

eliminating data corresponding to a diagonal blocks block $A(\text{nbase}:\text{nbase}+\text{m}, \text{nbase}:\text{nbase}+\text{m})$, where nbase being a position of an upperleft element of the diagonal block and m are integers integers, from data of the narrow blocks placed in each node, at each node;

redundantly allocating same data as the diagonal block which is eliminated at each node to each node commonly;

applying LU decomposition to both the diagonal block and ~~the~~ an allocated block by the assigning step in parallel in each node; and

updating the blocks of the matrix which is not yet LU-decomposed, using the LU-decomposed block, at each node~~[[.]]~~.

thereby realizing fast LU decomposition of a matrix effectively using a hardware of the parallel computer.